These are general guidelines that is ideal to implement in Ab Initio projects involving development, maintenance, testing activities. These are tips collected from various other sources from the net as well as from expert Ab Initio developers.

Project access control - Checking In and Checking out practices

- Before "Checking In" any graphs make sure that it has been deployed successfully.
- Also before "Checking In" inform the ETL Admin about the same.
- To obtain the latest version of the graph "Check Out" from EME Data store.
- Before running a graph "Check Out" from EME Data store to your individual sand box. In case the graph is not present in the EME Data store "Check In" and then run it.
- The Abinitio Sand Box for all authorized users should be created only by the ETL Admin.
- Before creating graphs on the server ensure that the User-ID, Password in the EME Settings and the Run Settings are the same.
- Before modifying a graph ensure that it is locked to prevent any sharing conflicts. When you lock a graph you prevent other users modifying it at the same time. It is advisable that individual graphs are handled by separate users.
- Do not create any table in the target database. In case it is needed, ask the DBA to do so.

- Any database related activities and problems should be reported to the concerned DBA immediately.
- Before you need to modify any table in the target database inform the concerned DBA and get his approval.
- Do not change any of the environment variables. As these environment variables are global to all graphs they should not be tampered with. Only the ETL Admin has rights to set or modify the environment variables.

Good practices for project implementation

- While running a graph one may encounter errors. Hence maintain error logs for every error you come across. A consolidated, detailed error sheet should be maintained containing error related and resolution information of all users. This can be used for reference when facing similar errors later on. In case you have a database error contact the DBA immediately.
- Ensure that you are using the relevant dbc file in all your graphs.
- Always validate a graph before executing it and ensure that it validates successfully. Deploy the graph after successful validation.
- ab_project_setup.ksh should be executed on regular basis. Contact ETL Admin for further details.
- Before running a graph check whether the test parameters are valid.
- After implementing the desired modifications save and unlock the graph.

Handling run time related errors

- If you are testing a graph created by some one else contact the person who created the

graph or the person who made recent modifications to it. He will assist you or himself perform the needful.

- If the error encountered relates to an Admin settings problem contact the ETL Admin immediately.

- If you face a problem that you have not encountered and resolved before, look in to the consolidated error sheet and check to see whether that problem has been previously faced and resolved by any other user. You can also approach various online tech forums to get further input on the error.

Documentation practices

- Maintain documents regarding all the modifications performed on existing graphs or scripts.

- Maintain ETL design documents for all graphs created or modified. The documents should be modified accordingly if any changes are performed on the existing graphs.

- While testing any graph follow the testing rules as per the testing template. Maintain documents for all testing activities performed.

What is good about underlying tables

- Ensure that in all the graphs where we are using RDBMS tables as input, the join condition is on indexed columns. If not then ensure that indexes are created on the columns that are used in the join condition. This is very important because if indexes are absent then there would be full table scan thereby resulting in very poor performance. Before execution of any graph use Oracle's Explain Plan utility to find the execution path of query.

- Ensure that if there are indexes on target table, then they are dropped before running the graph and recreated after the graph is run.

- If possible try to shift the sorting or aggregating of data to the source tables (provided you are using RDBMS as a source and not a flat file). SQL order by or group by clause will be much faster than Ab Initio because invariably the database server would be more powerful than Ab Initio server (even otherwise SQL order by or group by is done efficiently (compared to any ETL tool) because Oracle runs the statement in optimal mode.

- Bitmap indexes may not be created on tables that are updated frequently. Bitmap indexes tend to occupy a lot of disk space. Instead a normal index (B-tree index) may be created.

DML & XFR Usage

- Do not embed the DML if it belongs to a landed file or if it is going to be reused in another graph. Create DML files and specify as path.

- Do not embed the XFR if it is going to be re-used in another graph. Create XFR files and specify as path.

Efficient usage of components

- Skinny the file, if the source file contains more data elements than what you need for down stream processing. Add a Reformat as your first component to eliminate any data elements that are not needed for down stream processing.
- Apply any filter criteria as early in the flow as possible. This will reduce the number of records you will need to process early in the flow.
- Apply any Rollup's early in the flow as possible. This will reduce the number of records you will need to process early in the flow.
- Separate out the functionality between components. If you need to perform a reformat and filter on some data, use a reformat component and a filter component. Do not perform Reformat and filter in the same component. If you have a justifiable reason to merge functionality then specify the same in component description.